УДК 681.3

О.М. Данильченко, к.т.н., доц.
А.О. Данильченко, магістр
С.А. Ібрагім, аспір.
Житомирський державний технологічний університет

ЗАСТОСУВАННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ ПОБУДОВИ НАВЧАЛЬНИХ РОЗКЛАДІВ НА КЛАСТЕРНИХ СИСТЕМАХ

Сформульована задача складання тижневого розкладу ВНЗ. Обмеженнями задачі є основні вимоги до розкладу, а критерієм оптимальності – мінімум дефекту якості, який виникає завдяки порушенню неосновних вимог. Наведено генетичний алгоритм пошуку оптимального варіанта. Для збільшення швидкодії розроблена паралельна реалізація алгоритму на багатопроцесорній кластерній системі.

1. Постановка задачі складання розкладу вищого навчального закладу (СРВНЗу) та методи її рішення

Задача складання розкладу занять – одна з найбільш трудомістких і відповідальних у плануванні навчального процесу. Надалі ми будемо розглядати тижневий розклад як доданок семестрового диференційованого розкладу занять, що дозволяє забезпечити досить великий перелік пропонованих до нього методичних, організаційних та психофізіологічних вимог [1].

У загальному випадку, задачу складання розкладу навчальних занять можна сформулювати в такий спосіб: "Для заданого набору навчальних аудиторій (у даному випадку під навчальною аудиторією розуміється широке коло приміщень, у яких проводяться навчальні заняття (від комп'ютерної аудиторії до спортивного залу)) і заданого набору часових інтервалів (уроків чи навчальних пар) побудувати такий розподіл навчальних занять для всіх об'єктів (викладачі та навчальні групи), для якого обраний критерій оптимальності є найкращим".

До рішення задачі відомо велике число підходів. Однак її складність та відсутність практики кількісної оцінки є основними труднощами при формалізації й розробці ефективних алгоритмів.

У [2] задача складання розкладу розглядається з позиції теорії масового обслуговування; пропонується комбінаторний метод складання розкладів, заснований на властивостях перестановок. На цих властивостях побудований і алгоритм [3]. Алгоритми [2], [3] забезпечують розподіл однотипних занять на вільному полі розкладу з урахуванням обов'язкових вимог несуперечності розкладу. Однак вони не можуть бути використані для складання розкладу із заняттями різної тривалості.

Алгоритм у [4] здійснює покрокове конструювання рішення і використовує визначені евристичні вирішальні правила, засновані на застосуванні функцій переваги.

Підхід до побудови розкладів занять у [5] базується на теорії розкладів [6]. Він дозволяє створювати програмні генератори припустимих розкладів, що використовують ідеї послідовного конструювання, аналізу й відсівання варіантів.

У [7] запропоновані алгоритми, засновані на теорії графів; уперше сформульовані й доведені необхідні й достатні умови існування розкладів із заняттями різної тривалості.

Відомі спроби рішення задачі складання розкладу занять методами лінійного програмування [8]. У цьому випадку виникає ряд проблем, пов'язаних із складанням цільової функції та великою кількістю змінних.

Аналіз робіт [2]–[8] приводить до висновку, що відомі алгоритми дозволяють одержувати тільки припустимі розклади занять і не гарантують їхньої оптимальності.

У статті ленінградських авторів [11] пропонується для одержання оптимального тижневого розкладу занять використовувати метод гілок та меж. Однак, у цьому випадку, для одержання оптимального рішення задачі складання розкладу навчального закладу вимагаються невиправдано великі витрати часу.

Сформулюємо основні обмеження та критерій оптимальності для поставленої задачі.

Усі вимоги за важливістю можна розділити на основні й неосновні. Повне виконання всіх неосновних вимог до тижневого розкладу занять (при обов'язковому виконанні основних) найчастіше виявляється практично неможливим. У цьому випадку має місце дефект якості складеного розкладу.

 © О.М. Данильченко, А.О. Данильченко, С.А. Ібрагім, 2003

Виходячи з аналізу вимог, представляється розумним при побудові математичної моделі тижневого розкладу записати основні вимоги у вигляді обмежень задачі, а неосновні – звести в цільову функцію.

Таким чином, задача складання тижневого розкладу занять полягає в мінімізації дефекту якості, викликаного порушенням ряду неосновних вимог до розкладу занять при повному виконанні основних вимог.

За основні вимоги (обмеження) можна виділити наступні:

1. Проведення не більше одного заняття одним викладачем одночасно.
2. Непроведення в одному приміщенні більше одного заняття одночасно.
3. Неможливість проведення занять для викладача в заборонені тимчасові інтервали.
4. Проведення занять у спеціалізованих аудиторіях.
5. Неможливість проведення занять в аудиторіях у заборонені для них інтервали часу.

Виконання неосновних вимог бере на себе цільова функція задачі, яка дозволяє мінімізувати дефект якості розкладу, що складається. Кількісну оцінку дефекту якості розкладу будемо виражати сумою величин штрафів за кожен відступ від неосновних вимог.

До числа неосновних вимог варто віднести небажаність вільних пар (вікон) у груп студентів, проміжки часу між проведенням занять одним викладачем, проведення лекцій на початку й середині навчального дня та ін. Звичайно, перелік цих вимог не претендує на повноту і може мінятися з урахуванням специфіки навчального закладу.

Штраф за будь-яке порушення з переліку неосновних вимог до розкладу передбачається брати за десятибальною шкалою оцінок. При цьому розмір штрафу в кожному конкретному навчальному закладі можна попередньо установити за допомогою експертних оцінок.

2. Генетичний алгоритм рішення задачі СРВНЗу

Генетичні алгоритми – це алгоритми класу ймовірнісних алгоритмів рішення оптимізаційних задач. Уперше принципи генетичних алгоритмів були запропоновані Дж. Холландом у 1975 р. [9]. Генетичні алгоритми знаходять оптимальне рішення задачі з деякою ймовірністю і базуються на ідеї еволюції за допомогою природного добору, вперше описаного Ч. Дарвіном. Тому в генетичних (чи еволюційних) алгоритмах термінологія в основному аналогічна теорії еволюції Дарвіна.

Перше, з чим зіштовхується розроблювач власного генетичного алгоритму, – це кодування чи представлення рішення у вигляді хромосоми. В задачі СРВНЗу ми зіштовхуємося з деякими трійками (заняттями) виду – (предмет, викладач, група), яким варто знайти місце (аудиторію) і час виконання, тому як ген хромосоми пропонується використовувати такі трійки, а точніше їхні порядкові номери. У такий спосіб усі заняття нумеруються числами від 1 до N і хромосома являє собою деяку перестановку цих чисел. Порядок цих чисел у хромосомі показує, в якому порядку заняття будуть встановлюватися в розклад. Для побудови самого розкладу й обчислення функції придатності хромосом розроблена процедура, що буде поміщати заняття в розклад з урахуванням виконання всіх обов'язкових вимог і обчислювати функцію придатності з обліком усіх необов'язкових вимог. Процедура повертає велике число, якщо за даною хромосомою розклад не може бути побудовано, і суму штрафу розкладу, якщо він побудований.

Такий підхід дозволяє нам використовувати для рішення задачі СРВНЗу класичний генетичний алгоритм із високою швидкодією і перенести всю складність рішення та основний час на процедуру побудови розкладу й обчислення функції придатності для кожної хромосоми, що дасть нам виграш при реалізації паралельної версії алгоритму.

При розробці генетичного алгоритму його етапи й операції були реалізовані в такий спосіб:

– хромосоми в початковій популяції заповнювалися випадковими перестановками чисел від 1 до N;

– вибір хромосом для схрещування проводився за методом "турнірного добору";

– застосовувалося одномісне схрещування;

– як операція мутації використовувалася перестановка двох випадково обраних елементів у хромосомі;

– реалізовано регенераційний тип репродукції поколінь із переносом, при необхідності – кращої хромосоми з попереднього покоління в наступне;

– закінчення алгоритму проводилося або при розгляді заданої кількості поколінь, або при конвертуванні покоління.

У процесі знаходження рішення генетичним алгоритмом найбільш важливим є постійна підтримка допустимості рішень протягом роботи алгоритму, тобто підтримка хромосом такими, щоб вони не порушували обов'язкових обмежень. Це заощаджує процесорний час, що у противному випадку витрачався б на розрахунок і підтримку неприпустимих рішень. Оператор схрещування може вплинути на допустимість хромосом. Тому після виконання цього оператора потрібно перевіряти новостворену хромосому на наявність порушень обов'язкових обмежень і з появою таких виконати процедуру виправлення хромосоми. У даній реалізації алгоритму пропонується така процедура з обчислювальною складністю не більше ніж $O(N^2)$.

Оскільки будь-якому припустимому розкладу можна поставити у відповідність деяку перестановку чисел від 1 до N (причому одному розкладу може відповідати кілька перестановок), то, перебравши із деякою ймовірністю усі перестановки, ми переберемо всі припустимі розклади. За побудовою алгоритму значення найкращої функції придатності при переході від одного покоління до іншого не погіршується, тобто алгоритм розроблений правильно і, отже, при розгляді досить великої кількості поколінь ми з імовірністю, близькою до одиниці, одержимо оптимальне рішення.

3. Паралельна реалізація генетичного алгоритму СРВНЗу

Останнім часом в усьому світі бурхливо розвивається відносно нова область обчислювальної індустрії: метакомп'ютинг. Поняття метакомп'ютинг – це абстракція, за допомогою якої комп'ютерні кластери й окремі комп'ютери в розподіленій системі можуть бути представлені у вигляді єдиного віртуального комп'ютерного ресурсу. Під обчислювальним кластером мається на увазі масив, що складається з окремих машин, які взаємодіють через локальну мережу чи інший канал зв'язку і призначений для роботи паралельних додатків за моделлю передачі повідомлень [10].

У цьому випадку кожен вузол одержує доступ тільки до локальної пам'яті, і виконуваний паралельний алгоритм виявляється розділеним на ряд паралельних процесів, що обмінюються інформацією шляхом посилки/прийому повідомлень. Зазначений спосіб взаємодії здійснюється з використанням моделі передачі повідомлень, реалізованої у декількох різних стандартах, таких як MPI чи PVM.

Серед усіх операцій генетичного алгоритму найбільш тривалою є операція обчислення міри придатності чи оцінки кожної хромосоми. Тому було прийняте рішення розподілити між процесорами саме обчислення оцінок. Це було реалізовано шляхом поділу популяції на рівні частини і пересилання їх дочірнім процесам для подальшого обчислення оцінок. Основним завданням дочірнього процесу є одержання масиву хромосом, обчислення оцінок кожної хромосоми й повернення масиву цих оцінок батьківському процесу.

4. Обчислювальний експеримент та основні результати

На базі розробленого програмного забезпечення проведено обчислювальний експеримент, у результаті якого був отриманий графік залежності часу рішення задачі складання розкладів від числа процесів.

Як контрольний приклад були взяті дані за тижневим розкладом на двох факультетах. Варто помітити, що цей приклад носить скоріше перевірочний (контрольний) характер, а для складання реальних розкладів варто ще провести деяку роботу в плані уточнення навантаження викладачів, визначення штрафних коефіцієнтів для розкладів, визначення підмножин аудиторій для кожного предмета, обмежень на проведення занять тощо.

Приклад складався з наступного набору вихідних даних:

- кількість викладачів – 58;
- кількість предметів – 80;
- кількість груп – 65;
- кількість аудиторій – 37.

Розбиття по підгрупах не проводилось, що неістотно для нашого алгоритму, тому що при виникненні такої потреби як неподільну одиницю можна розглядати підгрупу, а практичні заняття для двох підгруп – як невелику лекцію.

У результаті розподілу навантаження, враховуючи те, що лекційні заняття проводяться для декількох груп, було отримано 1950 трійок (предмет, група, викладач). Розклад формувався на часовому інтервалі в 5 днів по 7 пар у кожному. В такий спосіб задача складання розкладу

зводиться до наступного. На поле 65*35 = 2275 варто розташувати 1950 занять з урахуванням обов'язкових обмежень.

Параметри генетичного алгоритму були обрані наступними:

- кількість хромосом у популяції – 200:
- кількість популяцій – 1000;
- коефіцієнт турнірного добору – 3;
- штрафний коефіцієнт вікон для груп – 5;
- штрафний коефіцієнт вікон для викладачів – 1;
- штрафні коефіцієнти небажаності проведення занять (наприклад лекцій) у визначений час – від 1 до 5.

Початкова популяція формувалася випадково. Для уточнення коефіцієнтів схрещування й мутації попередньо було розраховано близько 100 прикладів із невеликими наборами вихідних даних. В результаті було отримано, що з ростом коефіцієнта схрещування результати поліпшуються, а коефіцієнт мутації варто зменшувати. Найбільш сприятливими парами варто вважати коефіцієнт схрещування – 0.96–0.99, а мутації – 0.01–0.03.

Розкладу без штрафів одержати не вдалося (можливо, такого і не існувало). Найменше значення штрафу – 76 одиниць. Час розрахунку складав близько 17 годин. Причому обробка кожної популяції склала не набагато більше 1 хв. Основний час при розрахунку популяції витрачався на підрахунок функцій придатності – близько 0.9 хв. чи близько 0.27 сек. для кожної хромосоми.

Результати експерименту, а саме великий час розрахунку, підтвердили необхідність розробки паралельної версії алгоритму, тому що реальний приклад має бути в кілька разів більше за обсягом вихідних даних. Була підтверджена також ідея здійснювати розділення на паралельні процеси за даними при підрахунку функції придатності.

Тестування паралельної версії алгоритму проводилося на кластері ЖДТУ. Для скорочення часу розрахунків при тестуванні використовувався приклад із зменшеним у три рази обсягом даних. Час розрахунку без розпаралелювання склав 127 хвилин, із використанням двох процесів – 86 хвилин, трьох – 49. Усі результати представлені на рис. 1.

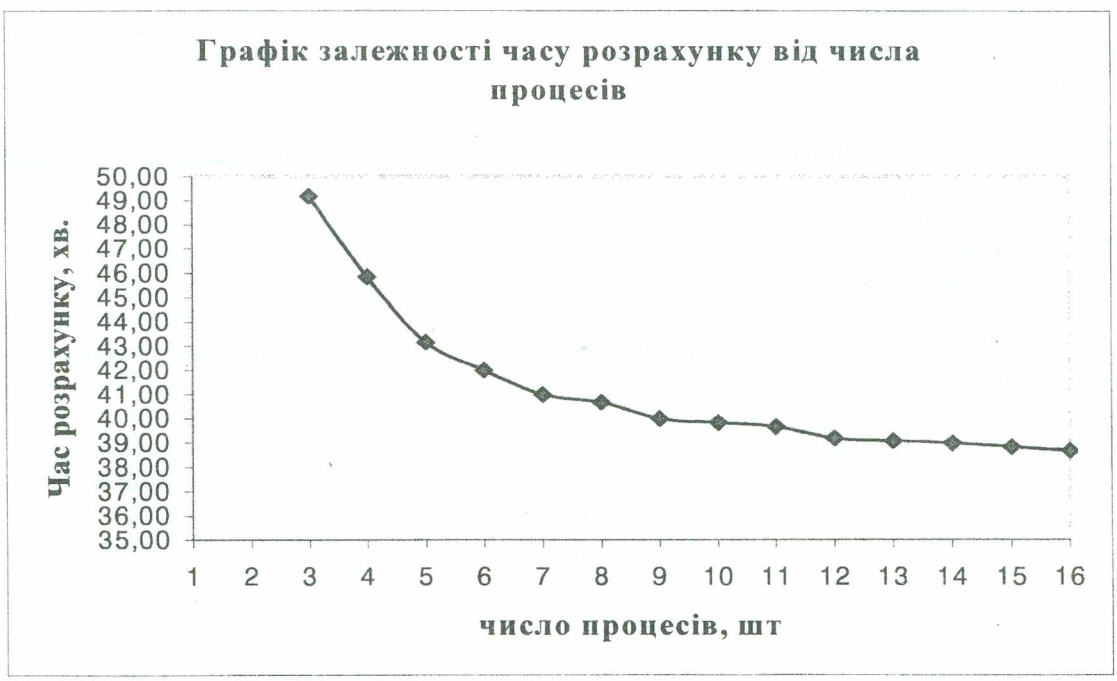

Рис. 1. Графік залежності часу розрахунку задачі від числа процесів

Слід зазначити, що найкращий результат – 38.7 хв., отриманий при використанні 16 процесів. Подальше збільшення кількості процесів не приводить до поліпшення результату через збільшення обсягу передачі даних між процесами.

ЛІТЕРАТУРА:

1. *Молибог А.Г.* Вопросы научной организации труда в высшей школе. – Минск: Высшая школа, 1975. – 292 с.
2. *Португал В.М.* Применение комбинаторного метода для решения задачи составления расписаний // Кибернетика. – 1967. – № 4. – С. 98–100.
3. *Николаев К.Г., Плужников Л.Н.* Алгоритм перестановок для построения допустимого расписания. – Изв. АН СССР. Техн. кибернетика. – 1969. – № 3. – С. 26–32.
4. *Анисимов Б.В. и др.* Применение ЭЦВМ для автоматизации процессов составления учебных планов и расписаний / В кн.: Использование ЭВМ в организации и планировании учебного процесса. – М., 1972. – С. 121–142.
5. *Баркан С.А., Танаев В.С.* О построении расписаний учебных занятий. – Изв. АН БССР. Сер. Физ.-мат. наук. – 1970. – № 1. – С. 76–82.
6. *Танаев В.С., Шурба В.В.* Введение в теорию расписаний. – М.: Наука, 1975. – 256 с.
7. *Krecrmar A.* Algorithm for constracting of university tametables and criterion of consistency of reguirements. – Acta cybernetica, 1973, 2. – № 2. – P. 136–145.
8. *Lawrie N. L.* Linear programming as an aid to school timetabling. – Britisch Joint Computer Conference. London, 1966.
9. *Holland John H.* Adaptation in natural and artificial systems. Ann Arbor The University of Michigan Press. – 1975. – 97 p.
10. *Воеводин В.В., Воеводин Вл.В.* Параллельные вычисления. – Санкт-Петербург.: БХВ-Петербург, 2002. – 599 с.
11. *Бабаев А.А., Друганов Б.Н.* Метод ветвей и границ в задаче составления недельного расписания занятий // Кибернетика. – 1981. – № 1. – С. 63–67.

ДАНИЛЬЧЕНКО Олександр Михайлович – кандидат технічних наук, доцент, завідувач кафедри програмного забезпечення обчислювальної техніки Житомирського державного технологічного університету.

Наукові інтереси:
- теорія розкладів;
- теорія складності екстремальних задач;
- паралельні обчислення.

ДАНИЛЬЧЕНКО Анна Олександрівна – магістр Житомирського державного технологічного університету.

Наукові інтереси:
- теорія розкладів;
- паралельні обчислення.

ІБРАГІМ Саад Алла – аспірант Житомирського державного технологічного університету.

Наукові інтереси:
- теорія розкладів;
- паралельні обчислення.

Подано 30.10.2003